

IBOX3588 Edge AI Embedded Industrial Computer



User Manual

Version: 1.5

© Vantron Technology, Inc. All rights reserved.

Revision History:

No.	Version	Description	Date
1	V1.0	First release	Sep. 26, 2022
2	V1.1	Added Ubuntu system manual, adjusted the typesetting accordingly	Nov. 28, 2022
3	V1.2	Updated certain I/O description	Feb. 13, 2023
4	V1.3	Updated serial port description as per the design change	May 18, 2023
5	V1.4	Aligned the serial port debugging commands of Debian system to the other systems	Dec. 8, 2023
6	V1.5	Separated the serial port debugging commands of Ubuntu and Debian systems from the Android system	Dec. 15, 2023

Table of Contents

Foreword	1
CHAPTER 1 INTRODUCTION.....	5
1.1 Product Overview.....	6
1.2 Unpacking	7
1.3 Specifications	8
1.4 Product Layout	9
1.5 Operating System	10
1.6 Mechanical Dimensions	10
1.7 Power Supply and Consumption	10
1.8 Environmental Specifications.....	10
CHAPTER 2 GETTING STARTED	11
2.1 Hardware Connection	12
2.2 Using the Device	15
2.3 Turning on/off the Device	16
CHAPTER 3 HARWARE DESCRIPTION.....	17
3.1 Power input.....	18
3.2 ON/OFF button.....	18
3.3 Ethernet jack	19
3.4 USB 2.0 + Ethernet jack.....	19
3.5 USB 3.0	19
3.6 USB Type-C.....	20
3.7 HDMI port	20
3.8 Micro SD & Micro SIM slots	20
3.9 Serial connectors.....	20
3.10 CAN	21
3.11 GPIO	22
3.12 M.2 slots.....	22
3.13 Antenna connectors	22
CHAPTER 4 FIRST-USE DEBUGGING.....	23
4.1 Serial connectors.....	24
4.1.1 Android system	25
4.1.2 Ubuntu and Debian systems	27
4.2 CAN	29
4.3 GPIO	31
4.3.1 Android system	31
4.3.2 Ubuntu and Debian systems	32
4.4 Watchdog Timer.....	33
4.5 Audio Play	34
CHAPTER 5 ANDROID SYSTEM MANUAL	37
5.1 Enable Developer Options	38

5.2	ADB Setup on the Windows Host.....	39
5.3	App Installation via ADB Commands.....	41
5.4	Firmware Upgrade on a Windows Host	42
CHAPTER 6 DEBIAN SYSTEM MANUAL		45
6.1	About the System.....	46
6.1.1	User and password.....	46
6.1.2	System information.....	46
6.2	System Settings	47
6.2.1	Language setup	48
6.2.2	Time and date setup	50
6.2.3	Network Configuration.....	51
6.3	Image Flashing on an Ubuntu Host	52
6.3.1	Prerequisites	52
6.3.2	Image flashing	52
CHAPTER 7 UBUNTU SYSTEM MANUAL		54
7.1	About the System.....	55
7.1.1	User and password.....	55
7.1.2	System information.....	55
7.2	System Settings	56
7.2.1	Language setup	56
7.2.2	Date & time setup.....	56
7.2.3	Network/Wi-Fi	56
7.2.4	User	56
7.3	Image Flashing on an Ubuntu Host	57
7.3.1	Prerequisites	57
7.3.2	Image flashing	57
7.4	ROS Tutorials	58
CHAPTER 8 DISPOSAL AND WARRANTY		59
8.1	Disposal	60
8.2	Warranty	61
Appendix Regulatory Compliance Statement		62

Foreword

Thank you for purchasing IBOX3588 Edge AI Embedded Industrial Computer (“the Device” or “the Product”). This manual intends to provide guidance and assistance necessary on setting up, operating or maintaining the Product. Please read this manual and make sure you understand the structure and functionality of the Product before putting it into use.

Intended Users

This manual is intended for:

- Operator of the Product
- Technical support engineers
- Other users

Copyright

Vantron Technology, Inc. (“Vantron”) reserves all rights of this manual, including the right to change the content, form, product features, and specifications contained herein at any time without prior notice. An up-to-date version of this manual is available at www.vantrontech.com.

The trademarks in this manual, registered or not, are properties of their respective owners. Under no circumstances shall any part of this user manual be copied, reproduced, translated, or sold. This manual is not intended to be altered or used for other purposes unless otherwise permitted in writing by Vantron. Vantron reserves the right of all publicly released copies of this manual.

Disclaimer

While all information contained herein has been carefully checked to assure its accuracy in technical details and typography, Vantron does not assume any responsibility resulting from any error or features of this manual, nor from improper uses of this manual or the software.

It is our practice to change part numbers when published ratings or features are changed, or when significant construction changes are made. However, some specifications of the Product may be changed without notice.

Technical Support and Assistance

Should you have any question about the Product that is not covered in this manual, contact your sales representative for solution. Please contain the following information in your question:

- Product name and PO number;
- Complete description of the problem;
- Error message you received, if any.

Vantron Technology, Inc.

Address: 48434 Milmont Drive, Fremont, CA 94538

Tel: (650) 422-3128

Email: sales@vantrontech.com

Regulatory Information



The Product is designed to comply with:

- FCC
- CCC
- UL

Please refer to the Appendix for Regulatory Compliance Statement.

Symbology

This manual uses the following signs to prompt users to pay special attention to relevant information.







	Caution for latent damage to system or harm to personnel
	Attention to important information or regulations

General Safety Instructions

The Product is supposed be installed by knowledgeable, skilled persons familiar with local and/or international electrical codes and regulations. For your safety and prevention of damage to the Product and other equipment connected to it, please read and observe carefully the following safety instructions prior to installation and operation. Keep this manual well for future reference.

- Do not disassemble or otherwise modify the Product. Such action may cause heat generation, ignition, electronic shock, or other damages including human injury, and may void your warranty.
- Keep the Product away from heat source, such as heater, heat dissipater, or engine casing.
- Do not insert foreign materials into any opening of the Product as it may cause the Product to malfunction or burn out.
- To ensure proper functioning and prevent overheating of the Product, do not cover or block the ventilation holes of the Product.
- Follow the installation instructions with the installation tools provided or recommended.
- The use or placement of the operation tools shall comply with the code of practice of such tools to avoid short circuit of the Product.
- Cut off the power before inspection of the Product to avoid human injury or product damage.

Precautions for Power Cables and Accessories

-  Use proper power source only. Make sure the supply voltage falls within the specified range. The Product is designed to use 9-36V DC. Always check whether the Product is DC powered before applying power.
-  Place the cables properly at places without extrusion hazards.
-  Use only approved antenna(s). Non-approved antenna(s) may produce spurious or excessive RF transmitting power which may violate FCC limits.
-  Cleaning instructions:
 - Power off before cleaning the Product
 - Do not use spray detergent
 - Clean with a damp cloth
 - Do not try to clean exposed electronic components unless with a dust collector
-  Power off and contact Vantron technical support engineer in case of the following faults:
 - The Product is damaged
 - The temperature is excessively high
 - Fault is still not solved after troubleshooting according to this manual
-  Do not use in combustible and explosive environment:
 - Keep away from combustible and explosive environment
 - Keep away from all energized circuits
 - Unauthorized removal of the enclosure from the device is not allowed
 - Do not replace components unless the power cable is unplugged
 - In some cases, the device may still have residual voltage even if the power cable is unplugged. Therefore, it is a must to remove and fully discharge the device before replacement of the components.

CHAPTER 1 INTRODUCTION

1.1 Product Overview

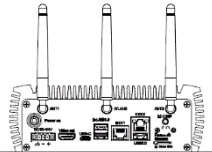




Vantron IBOX3588 Edge AI Embedded Industrial Computer is powered by Rockchip latest flagship RK3588 AIoT chipset that is equipped with an 8-core 64-bit CPU, an ARM Mali-G610 MP4 quad-core GPU, and a built-in AI acceleration NPU, capable of providing 6 TOPS computing power and supporting mainstream deep learning frameworks. With the development of the technology, there definitely will be a rising demand for AI-based products from the industrial control market, including but not limited to industrial robots, automated control, drones, etc.


The industrial computer offers two Gigabit Ethernet ports, supports 2.4GHz/5GHz Wi-Fi 6 and Bluetooth 5.1, and provides an M.2 slot for 4G/5G expansion to keep communication uninterrupted. It also supports 8K video decoding and encoding to deliver optimized display performance.

Since the industrial computer provides rich interfaces, a wide range of peripherals can be connected for extended applications like ARM PC, edge computing, cloud server, smart NVR, and other fields. Moreover, the different operating systems provide a stable and secure system environment for users.

1.2 Unpacking

The Product has been carefully packed with special attention to quality. However, should you find any component damaged or missing, please contact your sales representative in due time.

Standard accessories		Optional accessories	
	1 x IBOX3588 industrial computer		1 x 12V DC Power adapter & power cord
	2 x Wi-Fi & BT antenna		1 x DC power connector
\	\		2 x 4G LTE antenna / 4 x 5G antenna

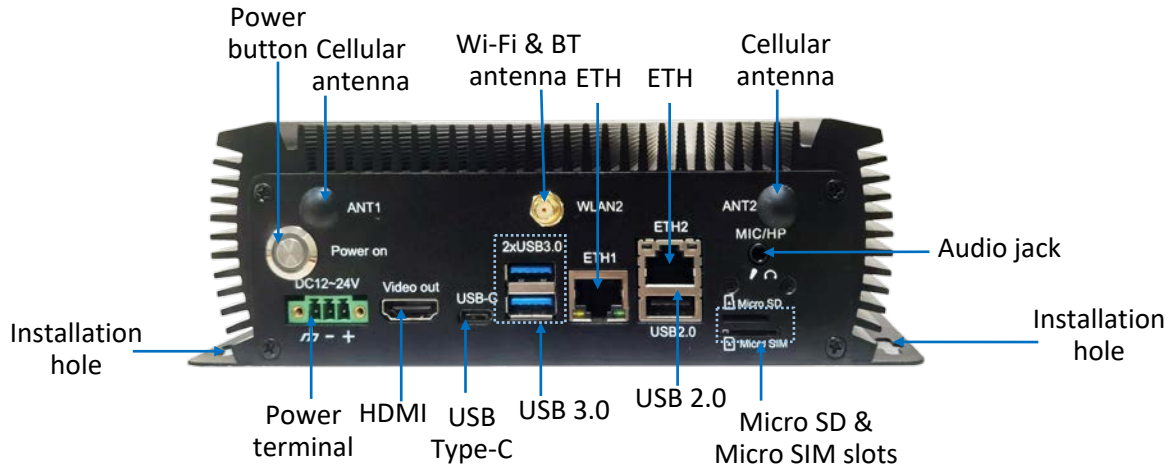
 *Actual accessories might vary slightly from the list above as the customer order might be different from the standard configuration options.*

1.3 Specifications

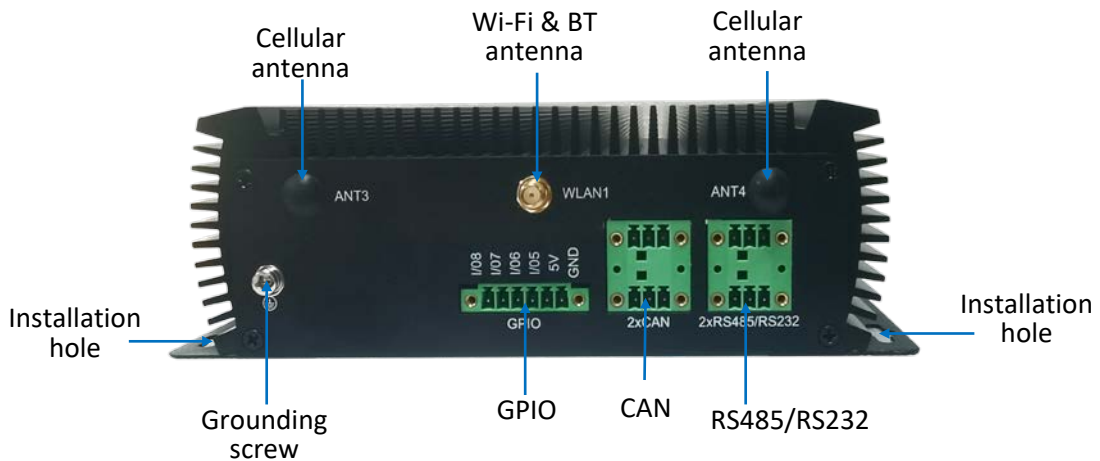
IBOX3588		
System	CPU	Rockchip RK3588, Quad-core Cortex-A76 + Quad-core Cortex-A55, 2.4GHz (Max.)
	GPU	ARM Mali-G610 MC4, 1GHz (Max.)
	NPU	6 TOPS
	Memory	8GB LPDDR4, up to 32GB
	Storage	32GB eMMC, up to 128GB SSD supported by an M.2 M-Key/M.2 B-Key (256GB ~ 1TB)
Communication	Ethernet	2 x RJ45, 10/100/1000Mbps
	Cellular	4G/5G (expansion by an M.2 B-Key)
	Wi-Fi & Bluetooth	Wi-Fi 802.11 a/b/g/n/ac/ax + BT 5.1
Media	Display	1 x HDMI (4096 x 2160 @60Hz)
	Video	1 x Video out port
	Audio	1 x Combo audio jack
I/Os	USB	2 x USB 3.0 Host, Type-A 1 x USB 2.0 Host, Type-A 1 x USB 3.0 Type-C (OTG supported)
	Serial port	2 x RS232/RS485 (2 x 3 x 3.81mm, Baud rate: 115200)
	SIM slot	1 x SIM slot
	Micro SD slot	1 x Micro SD slot
	RTC	Supported
Expansion	M.2 slot	1 x M.2 M-Key (2260/2280), PCIe 3.0 x 4, for SSD 1 x M.2 B-Key (2242/3052), USB 3.0/SATA 3.0, for 4G/5G/SSD
	GPIO	4 x GPIO on the Phoenix terminal
	CAN	2 x CAN on the Phoenix terminal
System Control	Button	1 x ON/OFF button (long press to reset)
	LED	1 x LED indicator for power and system status
Software	OS	Android 12, Debian, Ubuntu
	Language	English (default), Chinese
	Device management platform	BlueSphere MDM (Android version only)
	OTA tool	BlueSphere OTA
Power	Input	1 x Power distribution block (12V/5A, 24V/3A)
Mechanical	Dimensions	200mm x 143mm x 63.2mm
	Enclosure	Aluminum + sheet metal, black
	Weight	1200g
	Shock test	IEC 60068-2-27
	IP rating	IP40
Environment Condition	Temperature	Operating: -20°C ~ +70°C Storage: -40°C ~ +85°C
	Humidity	RH 0~95% (non-condensing)
	Certification	FCC, CCC, UL

1.4 Product Layout

1.4.1 Front view



1.4.2 Back view

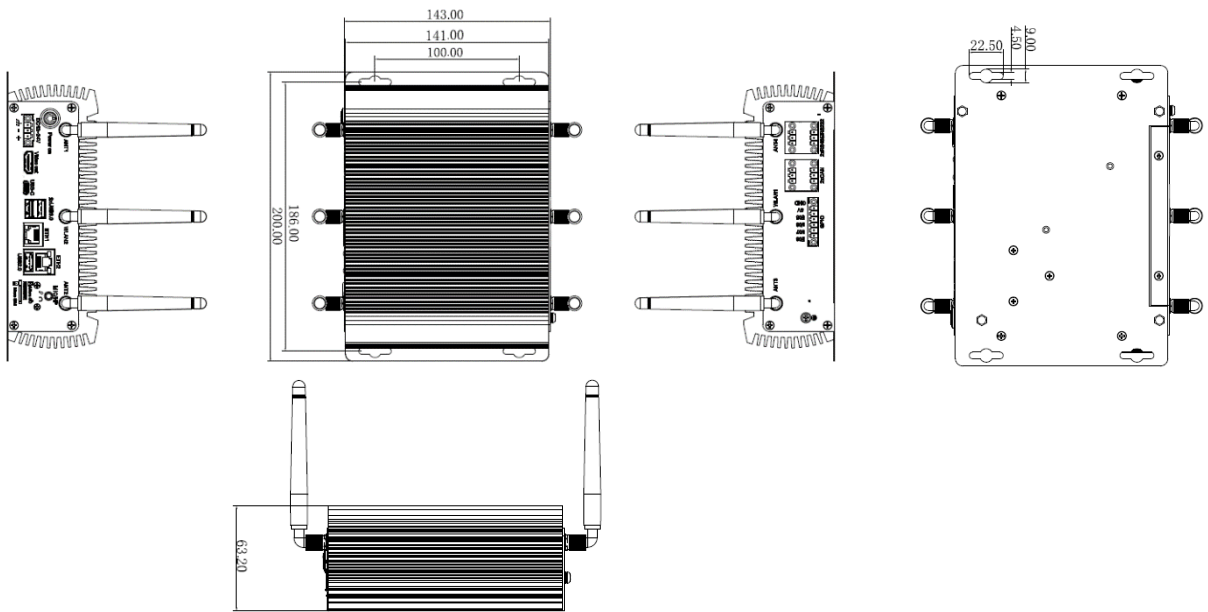


1.5 Operating System

The device supports multiples operating systems, such as Android, Debian, and Ubuntu.

1.6 Mechanical Dimensions

- 200mm x 143mm x 63.2mm



1.7 Power Supply and Consumption

IBOX3588 works with 12V/24V DC power input supplied by a DC connector.

The power consumption of the device is largely dependent on the RAM, storage capacity, peripherals, and other configurations of the Product.

1.8 Environmental Specifications

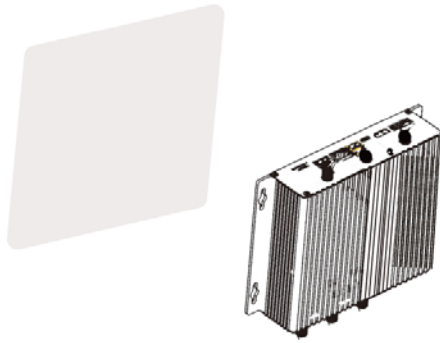
IBOX3588 works at a temperature ranging from 0°C to +60°C and at relative humidity of no more than 95% for non-condensing purpose. It is designed to be stored at a temperature ranging from -40°C to +85°C

CHAPTER 2 GETTING STARTED

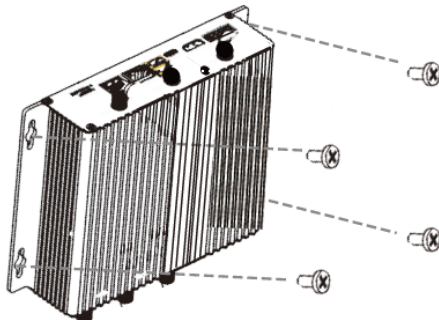
2.1 Hardware Connection

Before you proceed with the configuration of IBOX3588, follow the steps below to finish hardware connection.

1. Position the device on a desired place, for instance, a wall or a desktop;



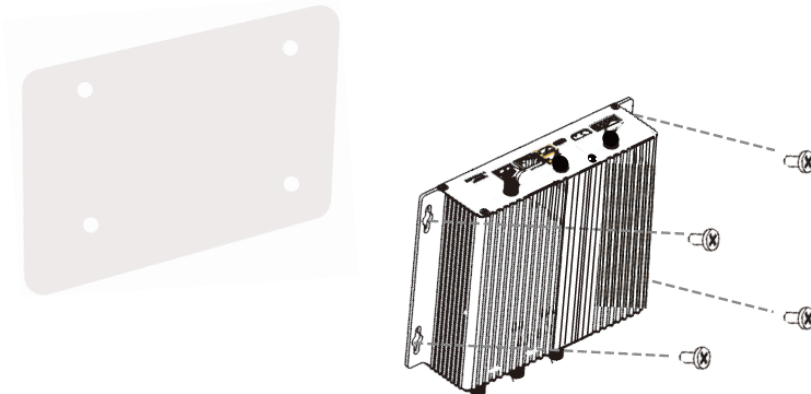
2. Align the screws (M4 x 6 / M4 x 8 recommended) with the screw holes reserved on the mounting bracket, and mark the position of the screws on the wall or the desktop;



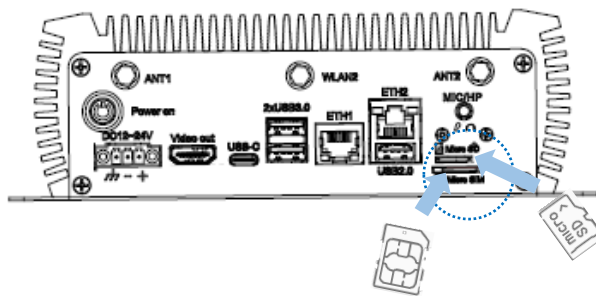
3. Drill four screw holes on the marks made in the prior step;



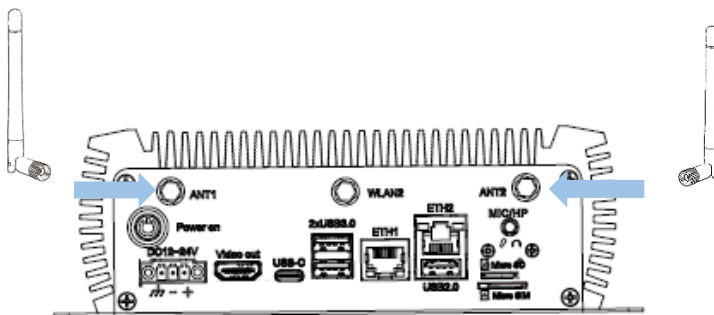
4. Place the device back and adjust the position of the bracket, if needed, then use a screwdriver to tighten the screws and secure the device;



5. Insert an active SIM card into the Micro SIM slot with the gold-colored contacts facing up and insert an SD card into the Micro SD slot with the gold-colored pins facing down;

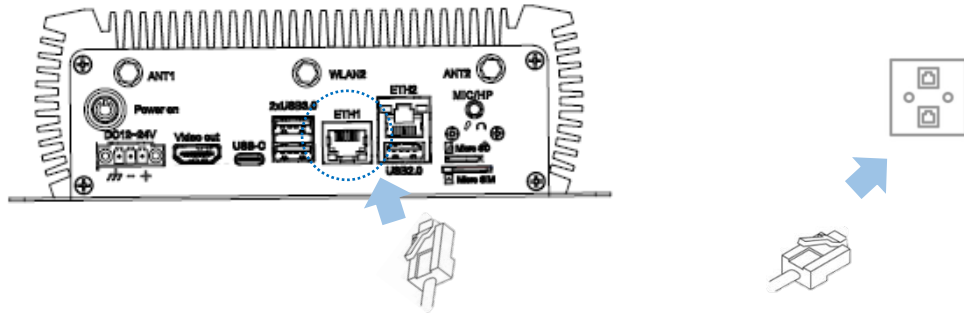


6. Install the antennas to the antenna connectors and tighten the connectors;

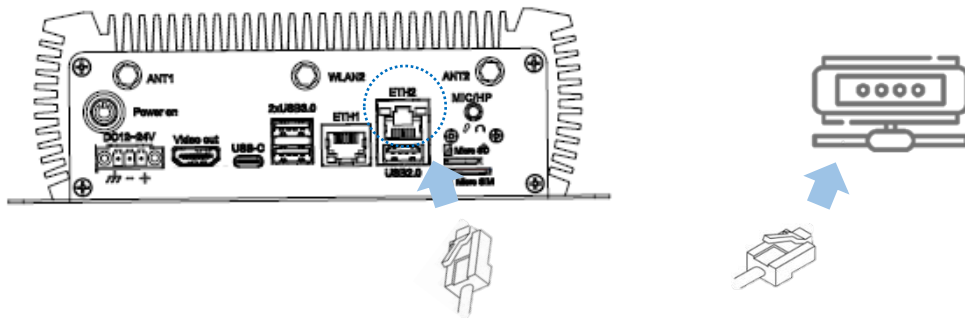


▶ There are no dedicated antenna connectors designed exclusively for 4G or 5G communication except that 4G communication requires two antennas, while 5G communication requires four.

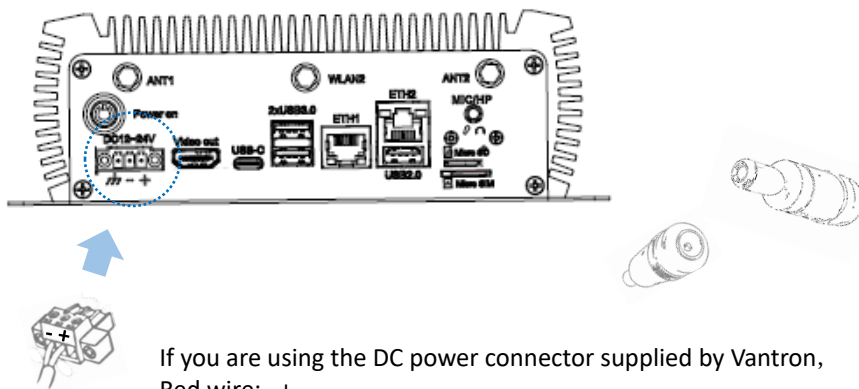
- When necessary, plug one end of an Ethernet cable into an Ethernet jack of IBOX3588 and the other to a live Ethernet port;



- When necessary, use another Ethernet cable to connect the other Ethernet jack of the device with a peripheral;



- Insert the terminal end of the female DC power connector into the power terminal of IBOX3588 and the other end into the power adapter;



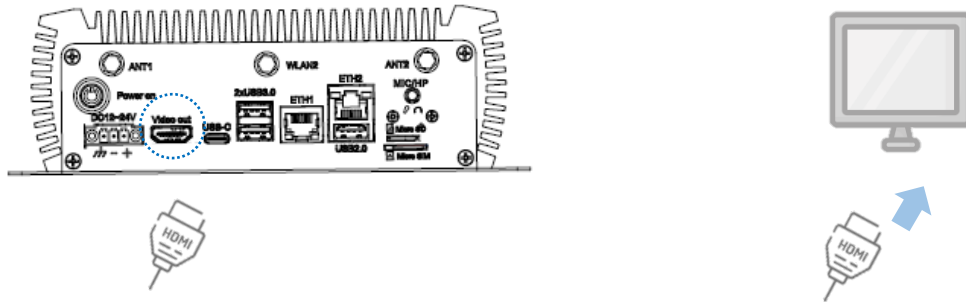
If you are using the DC power connector supplied by Vantron,
Red wire: +
Black wire: -

- Use a 12V/24V DC power adapter to power on IBOX3588;
- The LED indicator on the power button will turn solid green after power application.

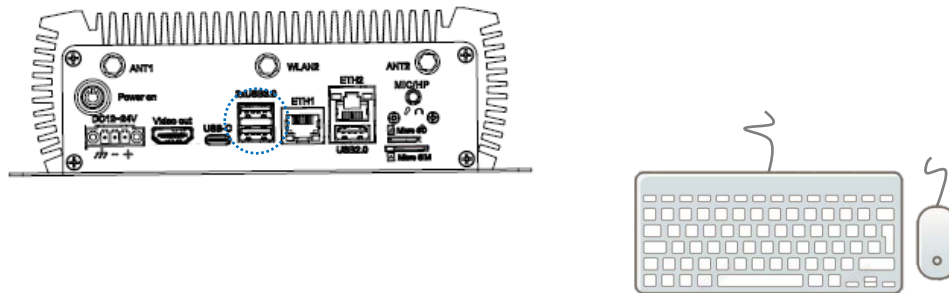
2.2 Using the Device

To access the GUI of the device, follow the steps below to connect it to a mouse, keyboard and monitor.

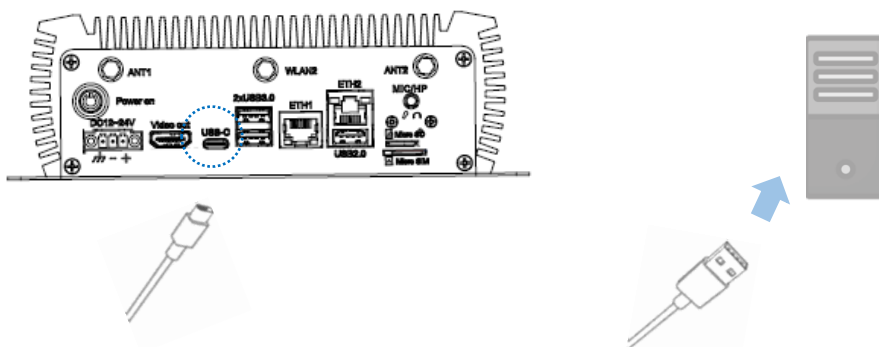
1. Insert one end of an HDMI cable to the HDMI interface of IBOX3588 and the other end to a monitor;



2. Connect a mouse and keyboard separately to the USB interfaces on the device;



3. Power on the device using a 12V/24 DC power adapter;
4. To debug the device remotely, use a USB Type-A to Type-C cable to connect the device to a host computer via the USB Type-C interface.



2.3 Turning on/off the Device

Having set up IBOX3588 according to the aforementioned steps, users can operate the device following the steps below:

- The device will automatically boot up after power-on;
- After IBOX3588 starts up, users can:
 1. Press the ON/OFF button once to turn off the monitor/wake up the device from the sleep mode;
 2. Press the ON/OFF button and hold for 2 seconds to call the power menu, and click **Power off** to turn off the device orderly;
 3. Press the ON/OFF button and hold for about 5 seconds to turn off the device directly;
 4. When necessary, unplug the power adapter from the power connector of IBOX3588 or the outlet to force shutdown.



The actions might vary with the operating system.

CHAPTER 3 HARWARE DESCRIPTION

This section briefs on the hardware definition and connector/jumper pinout.

3.1 Power input

IBOX3588 is designed to connect to a 12V/24V DC (12V 5A recommended) power supply with OVP/OCP protection.

Specification of the power connector: 1 x 3-pin, 3.81mm



Pin 1

Pinout description:

Pin	Signal	Description
1	GND	Ground
2	-VDC	Power input -
3	+VDC	Power input +

3.2 ON/OFF button



Button definition (Android OS):

Action (with power connected)	Description
Short press	Enable sleep mode or wake up the device Turn on the device if it is previously turned off
Press and hold for 1~2 seconds	Open the power menu to power off/restart the device/make an emergency call
Press and hold for 5 seconds	Power off

Button definition (Debian/Ubuntu OS):

Action (with power connected)	Description
Short press	Enable sleep mode or wake up the device
	Log out the account / Restart/shut down/lock the device
Press and hold for 1~2 seconds	Turn on the device
	Log out the account / Restart/shut down/lock the device
Press and hold for 5 seconds	Power off

There is an LED indicator on the ON/OFF button. When the device is connected to a live power source, the indicator will turn solid green.

3.3 Ethernet jack

IBOX3588 implements two Ethernet jacks.

Specification: RJ45, supporting 10M/100M/1000M Base-T4. Each Ethernet jack has two LED indicators to indicate the link/activity status of the network.

The following is ETH1.



3.4 USB 2.0 + Ethernet jack

The second Ethernet jack is integrated with a USB 2.0 Type-A port. The maximum output of the USB port is 5V/0.5A.



3.5 USB 3.0

IBOX3588 offers two USB 3.0 Type-A ports for connecting peripherals. Specification: Max. output 5V/1A.

3.6 USB Type-C

IBOX3588 implements a USB Type-C port that supports USB 3.0 OTG and DP 1.4 video output. Specification: Max. output 5V/1.5A.

3.7 HDMI port

IBOX3588 offers an HDMI port (Type-A), supporting HDMI 2.1 protocol with resolution up to 4096 x 2160 @60Hz.

3.8 Micro SD & Micro SIM slots

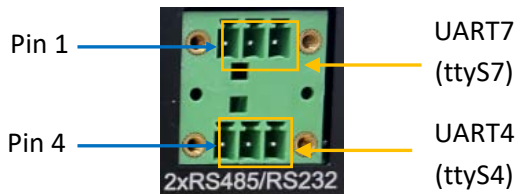
IBOX3588 provides a Micro SD card slot that supports a storage device of up to 128GB.

The SIM card slot supports SIM card hot plugging. The 4G LTE module is AT&T and Verizon pre-certified, and you may consult your sales executive for the module name before applying a SIM card from the carrier.

3.9 Serial connectors

IBOX3588 implements two RS232/RS485 multiplex connectors on a phoenix terminal.

The two multiplex connectors on the phoenix terminal are named as UART4 and UART7, corresponding to nodes /dev/ttyS7 and ttyS4, respectively.



Pinout description of RS232/RS485 connectors:

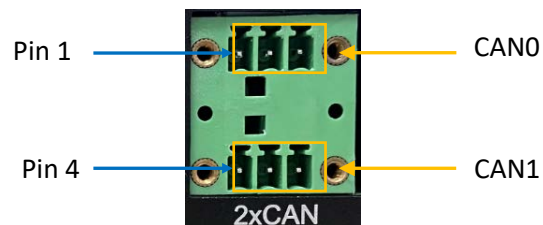
Pin	Signal	Description
1	RS485_A_2/ RS232_TX2	RS485 Bus I/O port, A / RS232 data output
2	RS485_B_2/ RS232_RX2	RS485 Bus I/O port, B / RS232 data input
3	ISO GND	Isolated ground
4	RS485_A_1/ RS232_TX1	RS485 Bus I/O port, A / RS232 data output
5	RS485_B_1/ RS232_RX1	RS485 Bus I/O port, B / RS232 data input
6	ISO GND	Isolated ground

Refer to **4.1** for serial connector function testing.

3.10 CAN

IBOX3588 implements two CAN connectors on a phoenix terminal.

Specification: 2 x 3 x 3.81mm.

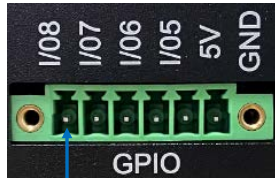


Pinout description:

Pin	Signal	Description
1	CAN0_H	High-level CAN0 bus line
2	CAN0_L	Low-level CAN0 bus line
3	GND	Ground
4	CAN1_H	High-level CAN1 bus line
5	CAN1_L	Low-level CAN1 bus line
6	GND	Ground

3.11 GPIO

IBOX3588 provides 4 GPIOs on a phoenix terminal to allow custom development of the device. Specification: 1 x 6 x 3.81mm.



Pin 1

Pinout description:

Pin	Signal	Description
1	GPIO8_5V	GPIO, 5V Level
2	GPIO7_5V	GPIO, 5V Level
3	GPIO6_5V	GPIO, 5V Level
4	GPIO5_5V	GPIO, 5V Level
5	VCC_5V0	5V power output
6	GND	Ground

3.12 M.2 slots

IBOX3588 offers two M.2 slots, including an M.2 M-Key slot (2260/2280) that supports PCIe 3.0 x4 for SSD expansion and an M.2 B-Key slot (2242/3052) that supports SATA 3.0 for SSD expansion or USB 3.0 for 4G/5G module connection.

3.13 Antenna connectors

The antenna connectors marked as ANT1, ANT2, ANT3, ANT4 on the enclosure are designed to connect 4G (optional two connectors) /5G antennas (four connectors). WLAN1 and WLAN2 antenna connectors are designed to connect Wi-Fi & Bluetooth antennas. There is no distinction between primary and diversity antennas.

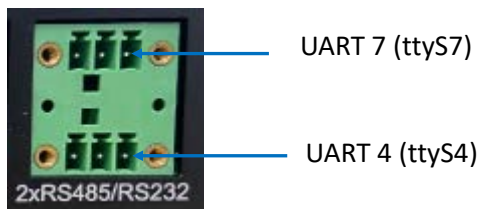
CHAPTER 4 FIRST-USE DEBUGGING

This chapter is mainly about the first-use debugging of interfaces and software applications. You can connect a key board, mouse and monitor to IBOX3588 industrial computer and debug the device directly in the console when it is running a Linux distribution, or, you can use a USB Type-A to Type-C cable to connect the device to a host computer for the same purpose.

4.1 Serial connectors

IBOX3588 offers two RS232/RS485 multiplexers on the Phoenix terminal, named as UART 7 and UART 4 which are mapped as /dev/ttyS7 and ttyS4 in the software system, respectively.

Refer to 3.9 for the pinout of the connectors.



You can use a proper USB-to-serial adapter to connect one serial port and the host computer first (TX-RX, RX-TX, GND-GND for RS232 wiring, A-A, B-B, GND-GND for RS485 wiring), then use a serial communication program to check if the serial port functions properly.

Alternatively, you can cross connect the two serial ports and test the functionality of the ports directly in the device console if the device is running Debian or Ubuntu system. If the device is running Android system, please remember to connect it to a host computer and execute the commands on the host computer.

4.1.1 Android system

The following is a summary of the commands for the two serial connectors in different modes when the device is running the Android system.

RS232 mode		
UART 4	Transmit data	# echo 0 > /sys/class/gpio/gpio497/value # echo 1 > /sys/class/gpio/gpio498/value # echo "Test string" > /dev/ttyS4
	Receive data	# echo 0 > /sys/class/gpio/gpio497/value # echo 1 > /sys/class/gpio/gpio498/value # cat /dev/ttyS4
UART 7	Transmit data	# echo 0 > /sys/class/gpio/gpio499/value # echo 1 > /sys/class/gpio/gpio500/value # echo "Test string" > /dev/ttyS7
	Receive data	# echo 0 > /sys/class/gpio/gpio499/value # echo 1 > /sys/class/gpio/gpio500/value # cat /dev/ttyS7

RS485 mode		
UART 4	Transmit data	# echo 0 > /sys/class/gpio/gpio497/value # echo 0 > /sys/class/gpio/gpio498/value # echo "Test string" > /dev/ttyS4
	Receive data	# echo 0 > /sys/class/gpio/gpio497/value # echo 0 > /sys/class/gpio/gpio498/value # cat /dev/ttyS4
UART 7	Transmit data	# echo 0 > /sys/class/gpio/gpio499/value # echo 0 > /sys/class/gpio/gpio500/value # echo "Test string" > /dev/ttyS7
	Receive data	# echo 0 > /sys/class/gpio/gpio499/value # echo 0 > /sys/class/gpio/gpio500/value # cat /dev/ttyS7

The example below illustrates how to test the functionality of the ports in RS232 mode:

1. Cross connect the serial connectors as shown below (TX-RX, RX-TX, GND-GND);



2. Open a terminal and use one port (UART 7 for instance) to receive data;

```
# echo 0 > /sys/class/gpio/gpio499/value  
# echo 1 > /sys/class/gpio/gpio500/value  
# cat /dev/ttyS7
```

3. Open another terminal and use the other serial port (UART 4) to transmit data;

```
# echo 0 > /sys/class/gpio/gpio497/value  
# echo 1 > /sys/class/gpio/gpio498/value  
# echo "Test string" > /dev/ttyS4
```

4. The first terminal will print the data received;
5. Refer to the tables on the previous page for commands specific to different modes and data transport purposes.

In this example, you can surely use single one serial port for transmitting and receiving the data.

4.1.2 Ubuntu and Debian systems

The following is a summary of the commands for the two serial connectors in different modes when the device is running the Ubuntu or Debian system.

RS232 mode		
UART 4	Transmit data	<pre># gpioset gpiochip7 4=0 # gpioset gpiochip7 5=1 # echo "Test string" > /dev/ttyS4</pre>
	Receive data	<pre># gpioset gpiochip7 4=0 # gpioset gpiochip7 5=1 # cat /dev/ttyS4</pre>
UART 7	Transmit data	<pre># gpioset gpiochip7 6=0 # gpioset gpiochip7 7=1 # echo "Test string" > /dev/ttyS7</pre>
	Receive data	<pre># gpioset gpiochip7 6=0 # gpioset gpiochip7 7=1 # cat /dev/ttyS7</pre>

RS485 mode		
UART 4	Transmit data	<pre># gpioset gpiochip7 4=1 # gpioset gpiochip7 5=0 # echo "Test string" > /dev/ttyS4</pre>
	Receive data	<pre># gpioset gpiochip7 4=0 # gpioset gpiochip7 5=0 # cat /dev/ttyS4</pre>
UART 7	Transmit data	<pre># gpioset gpiochip7 6=1 # gpioset gpiochip7 7=0 # echo "Test string" > /dev/ttyS7</pre>
	Receive data	<pre># gpioset gpiochip7 6=0 # gpioset gpiochip7 7=0 # cat /dev/ttyS7</pre>

The example below illustrates how to test the functionality of the ports in RS232 mode:

1. Cross connect the serial connectors as shown below (TX-RX, RX-TX, GND-GND);



2. Open a terminal and use one port (UART 7 for instance) to receive data;

```
# gpioset gpiochip7 6=0  
# gpioset gpiochip7 7=1  
# cat /dev/ttyS7
```

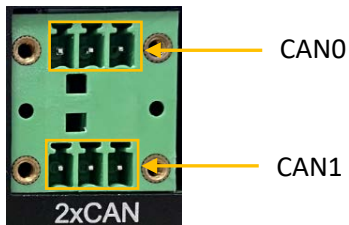
3. Open another terminal and use the other serial port (UART 4) to transmit data;

```
# gpioset gpiochip7 4=0  
# gpioset gpiochip7 5=1  
# echo "Test string" > /dev/ttyS4
```

4. The first terminal will print the data received;
5. Refer to the tables in the previous page for commands in different modes and different data transport purposes.

4.2 CAN

The device implements two CAN bus connectors (CAN0 and CAN1). You can cross connect the two connectors (H-H, L-L, GND-GND), and use one to receive data and the other to transmit data to test the functionality of the connectors. Refer to 3.10 for the pinout of the connectors.



- Use CAN0 to receive data and CAN1 to transmit data:
1. Open a terminal to set the baud rate of CAN0 and input the command to receive data;

```
# ip link set can0 up type can bitrate 125000  
# candump can0
```

2. Open another terminal to set the baud rate of CAN1 and input the command to send data.

```
# ip link set can1 up type can bitrate 125000  
# cansend can1 500#11.22.33.44
```

- Use CAN1 to receive data and CAN0 to transmit data:
1. Open a terminal to set the baud rate of CAN1 and input the command to receive data;

```
# ip link set can1 up type can bitrate 125000 (skip this if you have set the baud rate)  
# candump can1
```

2. Open another terminal to set the baud rate of CAN0 and input the command to send data.

```
# ip link set can0 up type can bitrate 125000 (skip this if you have set the baud rate)  
# cansend can0 500#11.22.33.44
```

- Check the status of CAN0:

```
# ip -details link show can0
```

```
root@linaro-alip:/# ip -details link show can0
2: can0: <NOARP,UP,LOWER_UP,ECHO> mtu 16 qdisc pfifo_fast state UP mode DEFAULT
group default qlen 10
    link/can promiscuity 0 minmtu 0 maxmtu 0
    can state ERROR-PASSIVE (berr-counter tx 0 rx 0) restart-ms 1
        bitrate 125000 sample-point 0.871
        tq 60 prop-seg 57 phase-seg1 57 phase-seg2 17 sjw 1
        rockchip_canfd: tseg1 1..128 tseg2 1..128 sjw 1..128 brp 1..256 brp-i
nc 2
        clock 99000000 numtxqueues 1 numrxqueues 1 gso_max_size 65536 gso_max
_segs 65535
root@linaro-alip:/#
```

- Check the status of CAN1:

```
# ip -details link show can1
```

4.3 GPIO

The device implements four GPIOs (GPIO5 ~ GPIO8), which are mapped as gpio489, gpio490, gpio491, gpio492 in the software system, respectively. You can refer to 3.11 for the pinout of the GPIOs.

Input the following command to check the status of the GPIOs:

```
# cat /sys/kernel/debug/gpio
```

The commands for setting the high/low level of the GPIO pins in Android system are different from those used in Ubuntu and Debian systems. You are recommended to set the high/low level of the GPIO pins with a reference to the commands for the respective system and use a voltmeter to check if the settings are valid.

4.3.1 Android system

The following table summarizes the commands for setting the high/low level of the GPIO pins in Android system.

GPIOs		
GPIO5	High level	# echo 1 > sys/class/gpio/gpio489/value
	Low level	# echo 0 > sys/class/gpio/gpio489/value
GPIO6	High level	# echo 1 > sys/class/gpio/gpio490/value
	Low level	# echo 0 > sys/class/gpio/gpio490/value
GPIO7	High level	# echo 1 > sys/class/gpio/gpio491/value
	Low level	# echo 0 > sys/class/gpio/gpio491/value
GPIO8	High level	# echo 1 > sys/class/gpio/gpio492/value
	Low level	# echo 0 > sys/class/gpio/gpio492/value

To switch the direction of the GPIOs:

```
# echo in > sys/class/gpio/gpioxxx/direction  
# echo out > sys/class/gpio/gpioxxx/direction
```

4.3.2 Ubuntu and Debian systems

The following table summarizes the commands for setting the high/low level of the GPIO pins in the Ubuntu or Debian system.

GPIOs		
GPIO5	High level	# gpioset gpiochip8 4=1
	Low level	# gpioset gpiochip8 4=0
GPIO6	High level	# gpioset gpiochip8 5=1
	Low level	# gpioset gpiochip8 5=0
GPIO7	High level	# gpioset gpiochip8 6=1
	Low level	# gpioset gpiochip8 6=0
GPIO8	High level	# gpioset gpiochip8 7=1
	Low level	# gpioset gpiochip8 7=0

4.4 Watchdog Timer

The watchdog timer is disabled by default. Once enabled, the watchdog timer is supposed to communicate with the system within every 44 seconds (dog feeding). If not, the system will reboot when the watchdog times out, indicating a system or application failure.

1. Write any character **except capital V** to open the watchdog;

```
# echo A > /dev/watchdog
```

2. Set the Runtime for dog feeding (e.g., 22 seconds) and the interval for system reboot upon an abnormality (e.g., 2 minutes);

```
[Manager]
LogLevel=warning
LogTarget=journal
#LogColor=yes
#LogLocation=no
#DumpCore=yes
#ShowStatus=yes
#CrashChangeVT=no
#CrashShell=no
#CrashReboot=no
#CtrlAltDelBurstAction=reboot-force
#CPUAffinity=1 2
#NUMAPolicy=default
#NUMAMask=
RuntimeWatchdogSec=22s
RebootWatchdogSec=2min
#ShutdownWatchdogSec=10min
#KExecWatchdogSec=0
#WatchdogDevice=
```

3. Disable the watchdog;

```
# echo V > /dev/watchdog
```

4. Users can also enable hardware watchdog in “systemd” and configure it likewise.

```
# /etc/systemd/system.conf
```

4.5 Audio Play

While both the HDMI and 3.5mm audio jack on the device support audio output, the audio jack further supports audio recording. For Android users, you can simply plug in a 3.5mm audio cable to use these functions. However, for Debian and Ubuntu users, you shall turn on the headphone manually to play the recorded audio clips.

Here are the commands to enable the audio recording and playing functions of the 3.5mm audio jack on Debian and Ubuntu systems.

1. List the audio cards on the device;

```
$ aplay -l
```

```
linaro@linaro-alip:~$ aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: rockchip-hdmi1 [rockchip-hdmi1], device 0: rockchip-hdmi1 i2s-hifi-0 [rockchip-hdmi1 i2s-hifi-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 1: rockchip-es8388 [rockchip-es8388], device 0: dailink-multicodecs ES8323.7-0011-0 [dailink-multicodecs ES8323.7-0011-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
linaro@linaro-alip:~$
```

HDMI interface

Audio jack

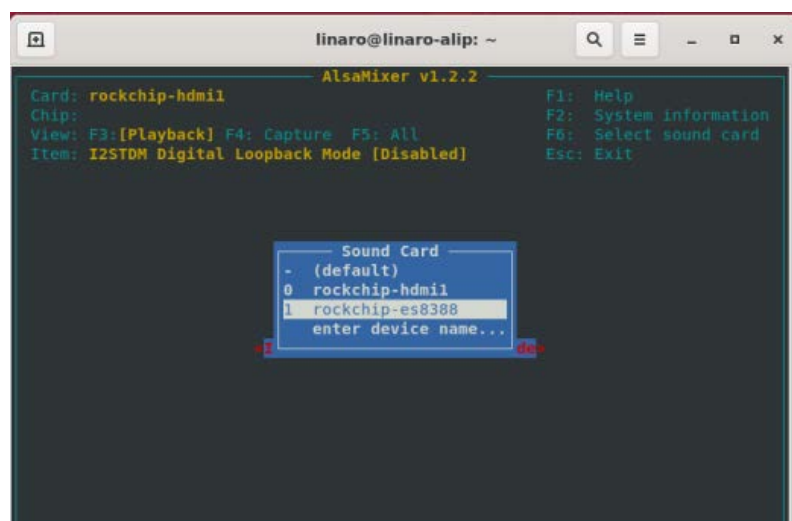
2. Insert the 3.5mm audio cable that supports both recording and playing functions into the audio jack;
3. Record an audio file and save it with a name such as 'test.wav';

```
$ arecord -D plughw:1,0 -f S16_LE -c 2 -r 44100 test.wav
```

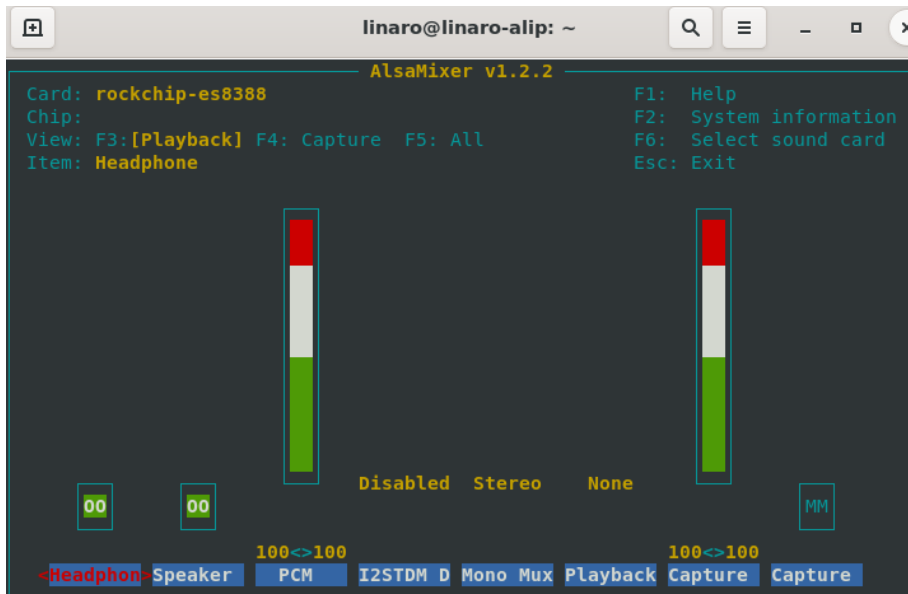
4. Open alsamixer program and press F6 to modify the audio settings;

```
$ alsamixer
```

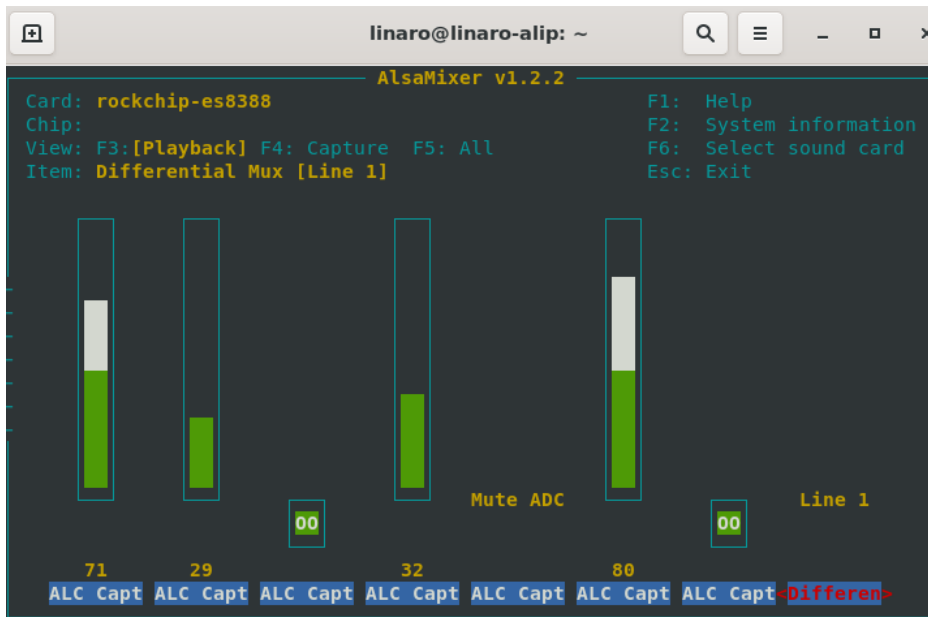
5. Use the up/down arrows on the keyboard to select '1 rockchip-es8388' and press Enter;



6. Use the left/right arrows on the keyboard to navigate to the <headphone> option, and press 'm' to enable the headphone function;



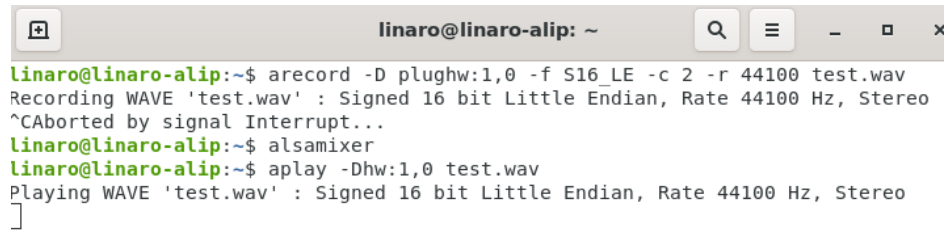
7. Use the left/right arrows on the keyboard to navigate to the <differential> option, and use the up/down arrows to select line 1;



8. Press ESC to exit the settings;

9. Input the following command to play the recorded audio.

```
Aplay -Dhw:1,0 test.wav
```

A terminal window titled 'linaro@linaro-alip: ~' with standard window controls. It shows a sequence of commands and their outputs: 'arecord' to record a test file, 'alsamixer' to check mixer levels, and 'aplay' to play the recorded file using hardware device 1,0.

```
linaro@linaro-alip:~$ arecord -D plughw:1,0 -f S16_LE -c 2 -r 44100 test.wav
Recording WAVE 'test.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
^CAborted by signal Interrupt...
linaro@linaro-alip:~$ alsamixer
linaro@linaro-alip:~$ aplay -Dhw:1,0 test.wav
Playing WAVE 'test.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
]
```

To switch to the HDMI and play the audio file from the interface, use the following command:

```
Aplay -Dhw:0,0
```


CHAPTER 5 ANDROID SYSTEM MANUAL

Prerequisites:

- IBOX3588
- A Windows host computer
- Software release package of IBOX3588
- A USB mouse, keyboard, and monitor for connecting the device for easier operation
- A 12V/24V power adapter for booting the device
- A USB Type-A to Type-C cable for connecting the device and the host computer

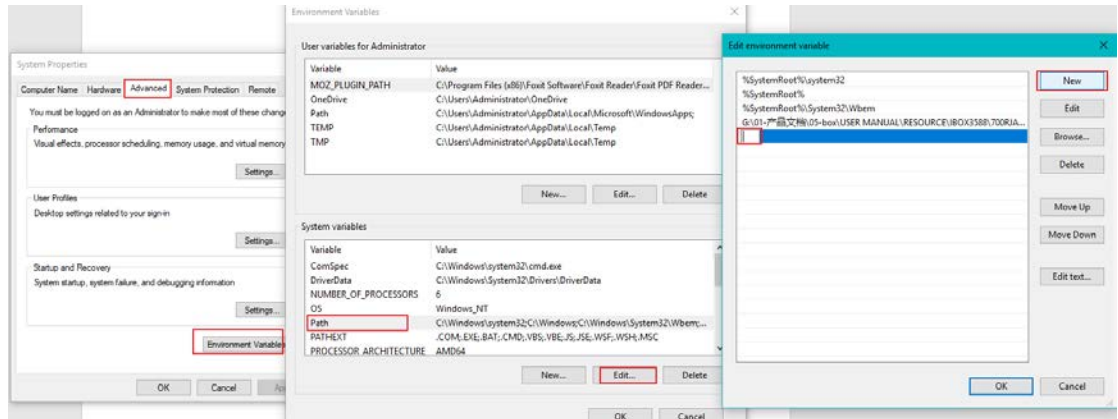
5.1 Enable Developer Options

To enable Developer options of IBOX3588, follow the steps below:

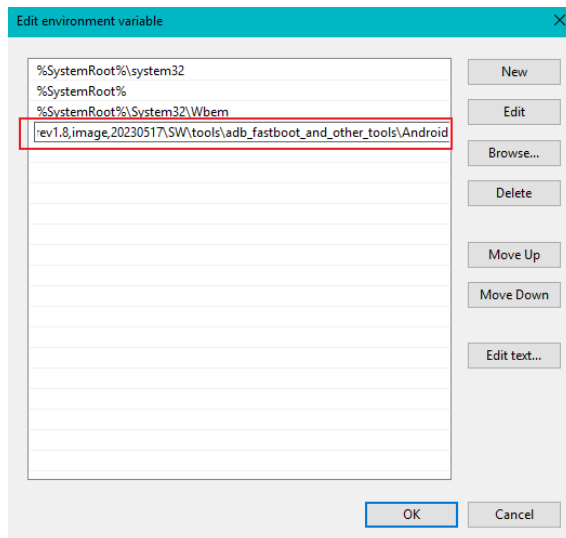
1. Connect IBOX3588 to a mouse, a keyboard, and a monitor for easier operations;
2. After the system has booted, swipe up to access the application drawer;
3. Click **Settings > About <tablet>** in sequence;
4. Scroll down to **Build number**, and click it consecutively for at least 7 times to enable **Developer options**;
5. Go back to **Settings > System > Advanced > Developer options** and toggle on **USB debugging**, then you can customize the device settings.

 Depending on the Android version, the entry might vary slightly.

- Click in sequence **Advanced > Environment Variables > System Variables > Path > Edit**, and click **New** in the pop-up;



- Paste the path of the **Android** folder, and click **OK** one by one to confirm and exit;



- Press "Win + R" and input **cmd** in the dialogue box;
- Input **adb version** in the command prompt to check if the ADB tool is installed.

```
Administrator: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.17763.1577]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>adb version
Android Debug Bridge version 1.0.31

C:\Users\Administrator>
```

5.3 App Installation via ADB Commands

In addition to the standard pre-installed Android applications, users can install their own applications on the device provided that it runs Android system and the ADB tool kit is installed and accessible on the Windows host computer.

1. Follow the steps in 5.2 to finish ADB setup on the host computer;
2. Connect the device and the host computer via a USB Type-A to Type-C cable;
3. Press “Win + R” and input `cmd` in the dialogue box;
4. Input `adb devices -l` in the command prompt to check if the device is connected to the host computer;

```
C:\Users\Administrator>adb devices -l
List of devices attached
874cc0aaede7d049    device product:occam model:Nexus_4 device:mako transport_id:1
```


5. When the device information is displayed under the command, you can copy the serial number (squared as shown above) for the next-step use;
6. Input the following command to install the app;


```
adb -s <serial number> install <APP path>
```


7. The installation will be executed after the command is input and the result of installation will be displayed below the command;

```
C:\Users\Administrator>adb -s 9fcfbfd4d37a6b4e install G:\tools\APKs\AnTuTuBenchmark_3400.apk
Performing Streamed Install
Success
```

8. The newly installed app will be displayed on the App drawer in the Alphabetic order.

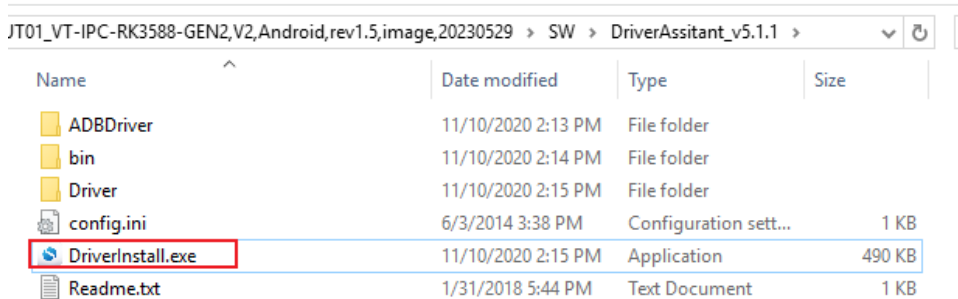
 *The screenshots are for illustration only and are not intended to represent the actual device number and other information of the device currently in your possession.*

 *In step 5 shown above, you can drag the .apk file from the local directory to the command line to replace the <App path> you typed in manually.*

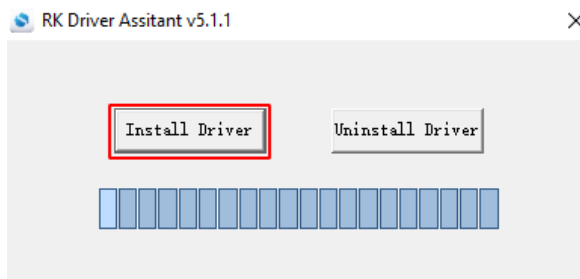
 *Make sure there is no special character in the app path. If you failed to install the apk, try using the absolute path of the .apk file enclosed in double quotation marks.*

5.4 Firmware Upgrade on a Windows Host

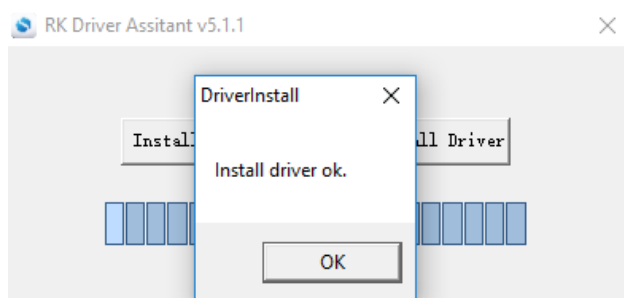
1. Follow the steps in 5.2 to finish ADB setup on the host computer;
2. Unzip the release package, and open the directory of the upgrade driver (\SW\DriverAssitant_vxxx), and locate the **DriverInstall.exe**;



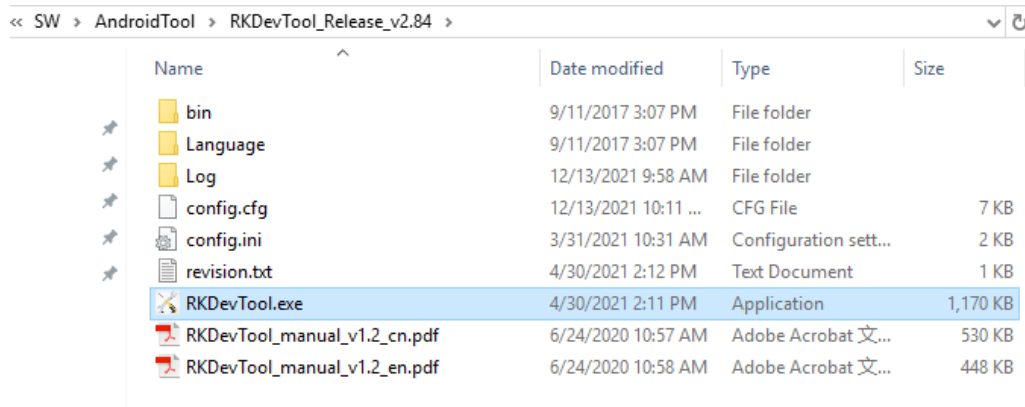
3. Right click the mouse and run the program as administrator;
4. Click **Uninstall Driver** first to remove the previously installed driver, if any, then click **Install Driver** and wait;



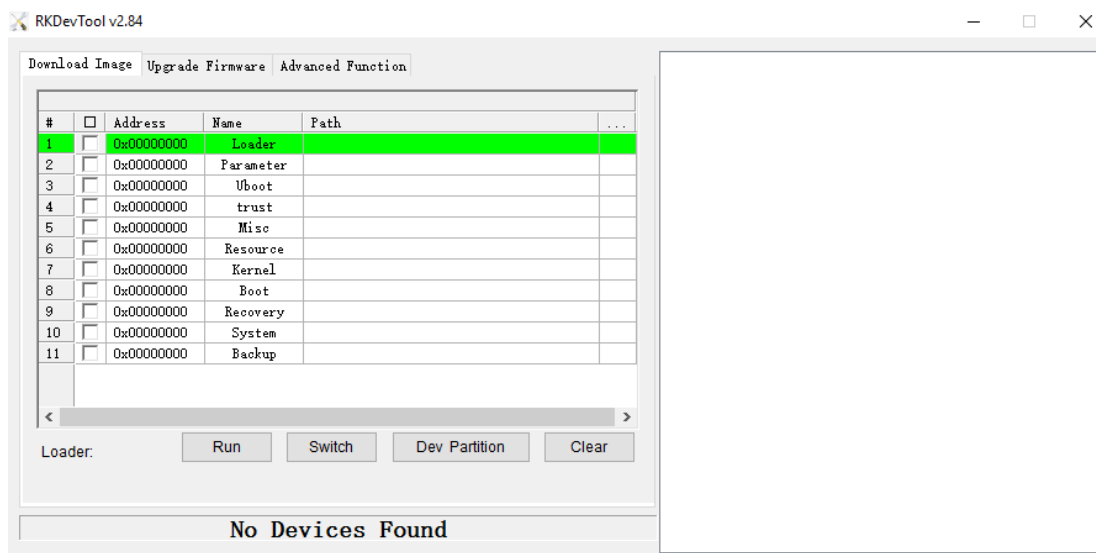
5. A pop-up will appear in a second suggesting the driver is installed;



- Return to the SW folder and open the directory of the upgrade tool (\\SW\\AndroidTool\\RKDevTool_Release_vxxx\\RKDevTool);

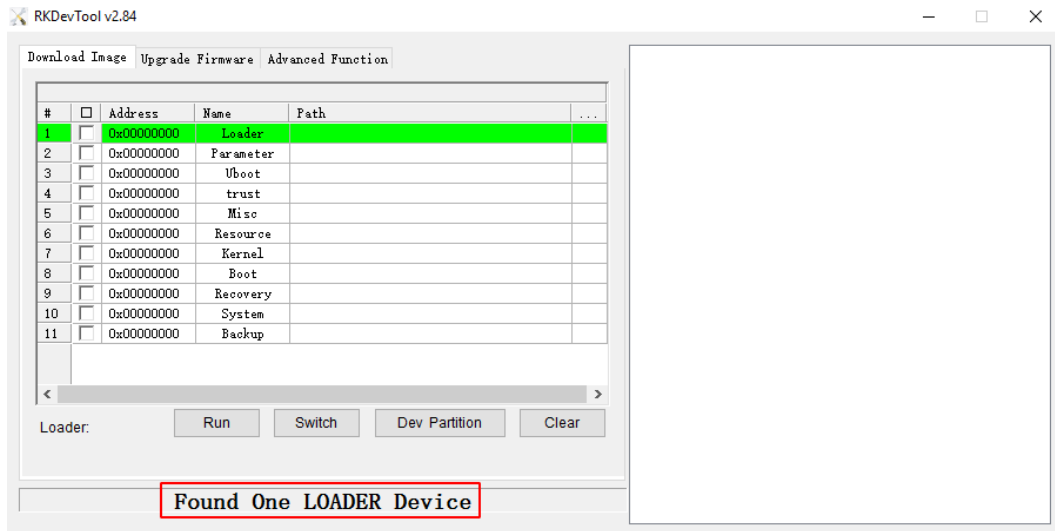


- Double click the upgrade tool program **RKDevTool.exe** to open the upgrade window;

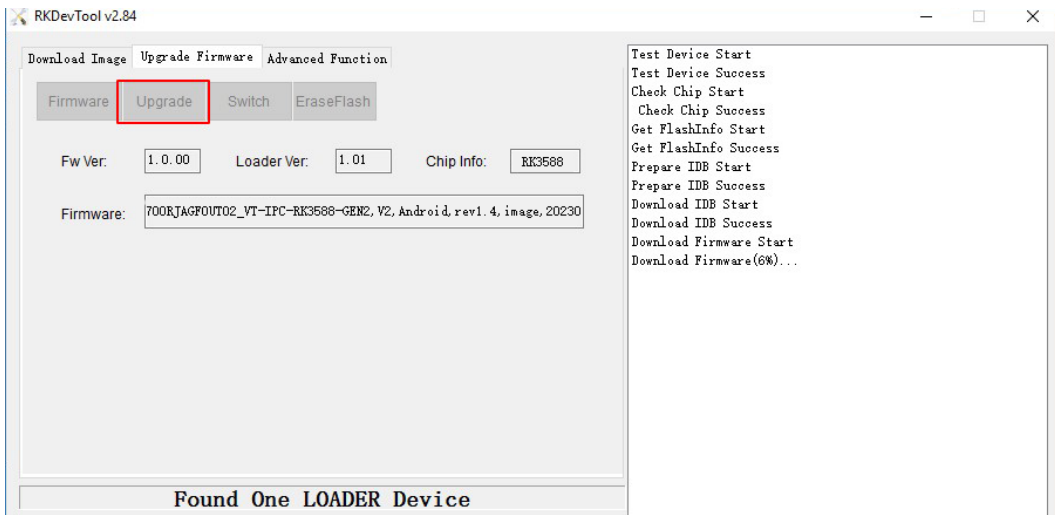


- Connect IBOX3588 to the Windows host computer using the USB Type-A to Type-C cable;
- Press Windows key + R and input `cmd` in the dialog box to open the command prompt;
- Input `adb devices` in the command prompt to check if the device is connected to the Windows host;
- Once the device is identified by the Windows host computer, input `adb reboot loader` to reboot the device into the bootloader mode;

12. Then the upgrade window will prompt for the existence of a Loader device, indicating that the upgrade process is ready;



13. Click **Upgrade Firmware > Firmware** in the upgrade window;
14. Open the image file (**update.img**) from the local directory (\SW\Image), and the firmware details will be automatically populated;
15. Click the **Upgrade** button and the system will start to download the image and upgrade the firmware automatically;



16. When the upgrade finishes, the system will reboot automatically.

CHAPTER 6 DEBIAN SYSTEM MANUAL

6.1 About the System

Debian is a Linux distribution and the device was built by Debian GNU/Linux 11 operating system.

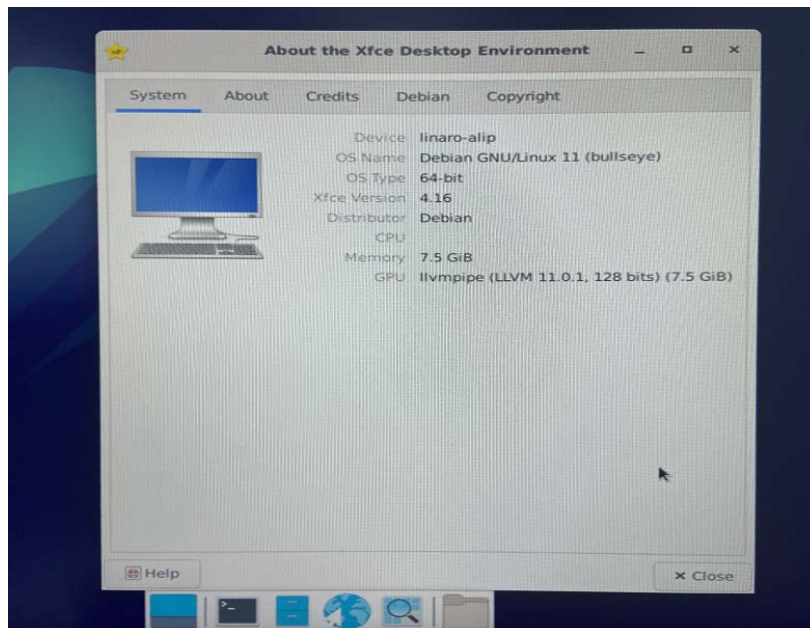
6.1.1 User and password

The system will automatically log in to **linaro** as the default user after bootup.

- User: linaro
- Password: linaro

6.1.2 System information

1. Connect a keyboard, a mouse and a monitor to the device for easy operation;
2. Power on the device and the system will log in to linaro as the default user;
3. Right click the mouse in an empty area to open the property dialogue of the device;
4. Click **Applications > About Xfce** to enter the desktop environment, then you can check the system information like device name, operating system, copyright statement, etc.



6.2 System Settings

By clicking on **linaro** on the top right corner of the screen, you can

- Lock the screen;
- Switch the user;
- Shut down/suspend the device;
- (Upon a click of the “Log out” option) Log out/restart/shut down/suspend the device or switch the user while saving the session for future logins.

The menu bar on the top of the screen allows the user to:

- Change the brightness and power settings of the device;
- Adjust the system volume and sound settings;
- Add the device to a Wi-Fi network and pair it with a Bluetooth device;
- Switch between different workspaces.

Other system settings are accessible from **Applications** on the top left corner of the screen or upon a right click of the mouse in an empty area of the screen.

6.2.1 Language setup

Assume the system language is Simplified Chinese. The following example demonstrates how to change the system language to English (US).

1. Open the Terminal from the bottom of the desktop and input `sudo su` to switch to the root mode;
2. Run the following command to check the language currently used by the system (the system uses `zh_CN.UTF-8` as the locale);

```
# env | grep LANG
```



```
linaro@linaro-alip: ~  
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)  
linaro@linaro-alip:~$ sudo su  
root@linaro-alip:/home/linaro# env | grep LANG  
LANGUAGE=zh_CN.UTF-8  
LANG=zh_CN.UTF-8  
root@linaro-alip:/home/linaro#
```

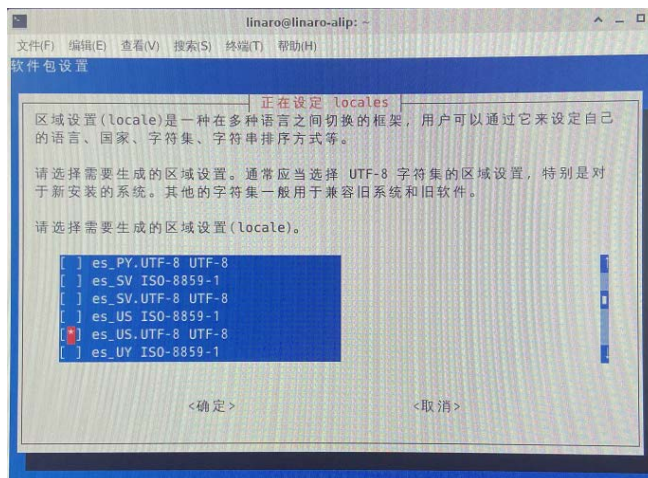
3. Run the following command to call the locale configuration page (`en_US.UTF-8` refers to the locale for the target language that you wish to switch to);

```
# export LANG=en_US.UTF-8
```

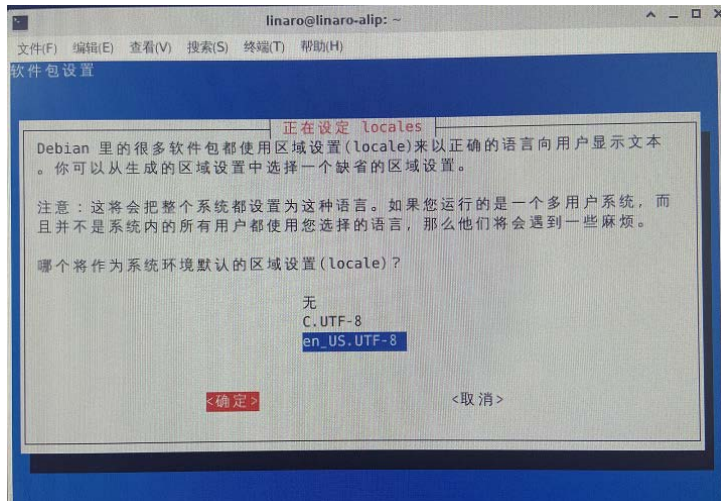
4. Use the following command to re-configure the locales;

```
# dpkg-reconfigure locales
```

5. Use the up and down arrows on the keyboard to move to the target locale (`en_US.UTF-8`);
6. Press “Space” in front of the target locale to select the language;



7. Move down to the current locale (zh_CN.UTF-8) and press "Space" to uncheck the current language;
8. Press "Tab" to move the cursor to the left option <OK> (确定 in Chinese) and press "Enter" to confirm your choice;
9. In the following pop-up, use the up and down arrows to move to the target locale (en_US.UTF-8) in the list, then press "Tab" to move the cursor to the left option <OK> (确定 in Chinese) on and press "Enter" to confirm your choice;



10. Wait a few seconds for the locale to take effect;

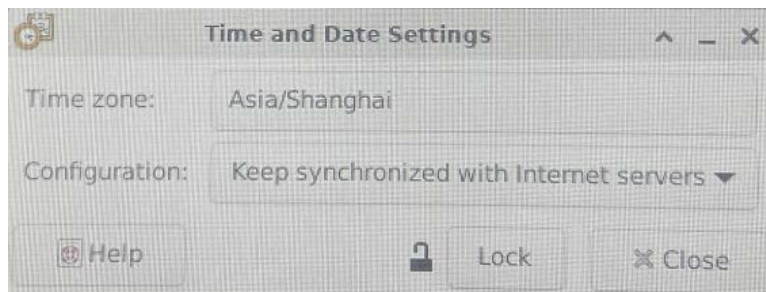
```
linaro@linaro-alip: ~  
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)  
linaro@linaro-alip:~$ sudo su  
root@linaro-alip:/home/linaro# env | grep LANG  
LANGUAGE=zh_CN.UTF-8  
LANG=zh_CN.UTF-8  
root@linaro-alip:/home/linaro# export LANG=en_US.UTF-8  
root@linaro-alip:/home/linaro# dpkg-reconfigure locales  
Generating locales (this might take a while)...  
  en_US.UTF-8... done  
Generation complete.  
root@linaro-alip:/home/linaro#
```

11. Then input **reboot** to restart the device;
12. Select "Update names" after device reboot to update the files to the current language;
13. Run **locale -a** command to check if the locale is set as the default system language.

6.2.2 Time and date setup

To change the system time and date, click **Applications** on the top left corner of the screen or right click the mouse in an empty area of the screen to call the menu:

1. Click **System > Time and Date** to open the Time and Date setup page;
2. Click **Unlock** and enter the password (linaro) before editing;
3. Click Time zone and select your own time zone from the list;
4. Choose to synchronize the time with the Internet server (device shall have Internet access) or fill in the time manually;



5. Click **Lock** to let the settings take effect.

6.2.3 Network Configuration

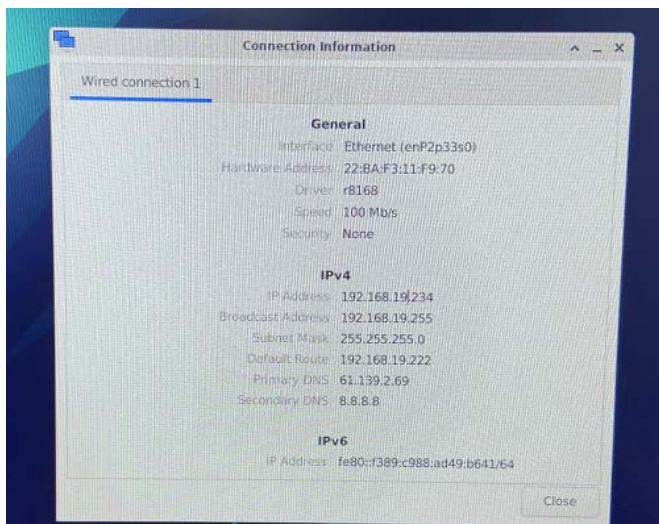
The device is configured to have Internet access once it connects to a live Ethernet or joins to a Wi-Fi network.

Make sure you have installed the Wi-Fi antennas before connecting to an existing wireless network.

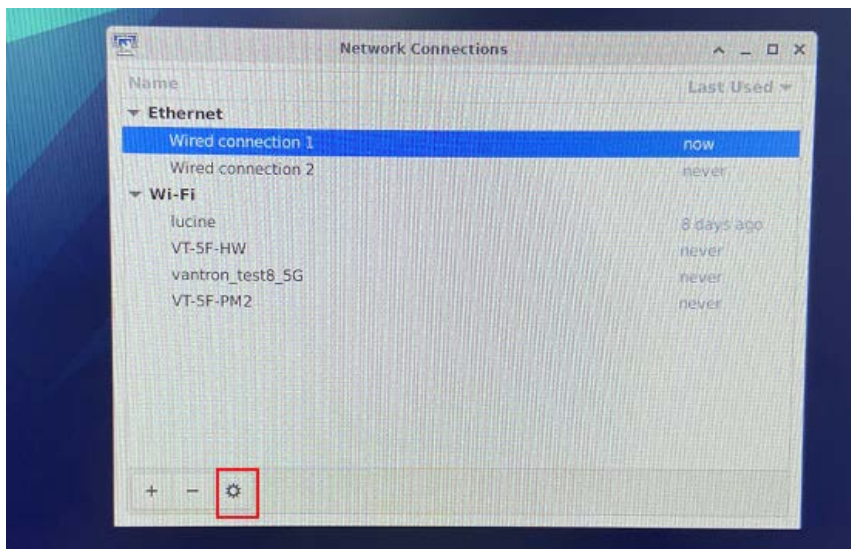
1. To join to a Wi-Fi network, click the network icon (dual-host/Wi-Fi icon) on the menu bar and search for the target network in the list;



2. To check the current network connection, you can move the cursor to the network icon, and right click the mouse to select **Connection Information**:



3. To edit the network configurations, you can right click the network icon and select **Edit Connections**, then select a connection point and click the cog icon to edit the network settings.



6.3 Image Flashing on an Ubuntu Host

6.3.1 Prerequisites

- IBOX3588
- An Ubuntu host (Ubuntu 18.04 or later recommended)
- A USB Type-A to Type-C cable
- Software release package of the device

6.3.2 Image flashing

1. Connect the device with the host computer via the USB Type-A to Type-C cable;
2. Open a terminal and input the following command line to install the ADB tool if necessary;

```
$ sudo apt-get install adb -y
```

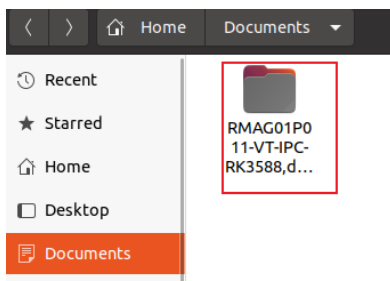
3. Check if the device is connected to the Ubuntu host via ADB;

```
$ adb devices -l
```

4. Run `adb shell` to execute shell commands on the device;
5. Enter `reboot loader`, and the device will reboot automatically and enter the bootloader mode;

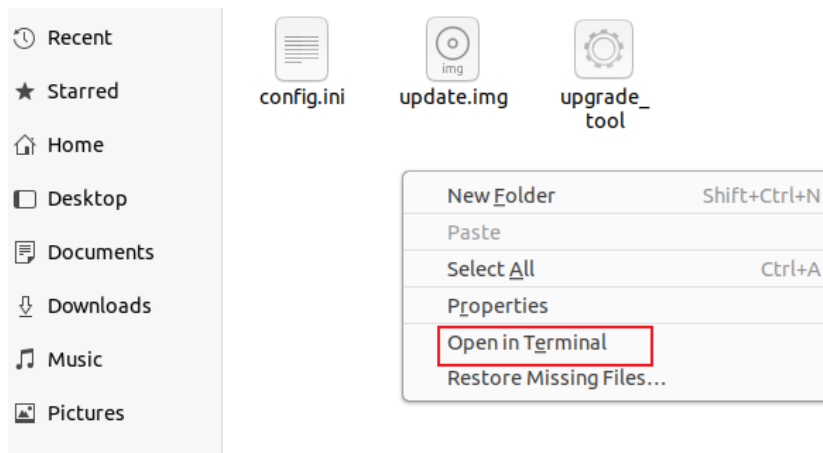
```
root@linaro-alip:~$ adb shell
root@linaro-alip:/# reboot loader
root@linaro-alip:/# @ubuntu:~$
```

6. Copy the image package provided by Vantron to the Ubuntu host (for instance, in the Documents folder);



7. Unzip the package and open the **img** folder;

8. Right click the mouse in an empty area and click **Open in Terminal** to execute the subsequent commands in the new terminal;



9. Input the following command line to start the upgrade process;

```
sudo ./upgrade_tool uf update.img
```

10. Input the password to download the system image;

```
@ubuntu:~/Documents/RMAG01P011-VT-IPC-RK3588,debian,V1.0.0,2022-08-01/img
$
sudo ./upgrade_tool uf update.img
[sudo] password for @ubuntu:
Using /home/@ubuntu/Documents/RMAG01P011-VT-IPC-RK3588,debian,V1.0.0,2022-08-01/
img/config.ini
Program Log will save in the /root/upgrade_tool/log/
Loading firmware...
Support Type:RK3588      FW Ver:1.0.00   FW Time:2022-07-30 16:22:22
Loader ver:1.0b Loader Time:2022-07-30 16:02:36
Download Image Total(5829691K),Current(2798535K)
```

11. The system will start upgrade once the download finishes, then the device will reboot automatically when the upgrade finishes.

```
Program Log will save in the /root/upgrade_tool/log/
Loading firmware...
Support Type:RK3588      FW Ver:1.0.00   FW Time:2022-07-30 16:22:22
Loader ver:1.0b Loader Time:2022-07-30 16:02:36
Upgrade firmware ok.
```

CHAPTER 7 UBUNTU SYSTEM MANUAL

7.1 About the System

Ubuntu is a Linux distribution and the device is running Ubuntu 20.04.LTS.

7.1.1 User and password


Users need to login to the system after system boot.

- User name: linaro
- Password: linaro

7.1.2 System information

1. Connect a keyboard, a mouse and a monitor to the device for easy operation;
2. Power on the device and input the password to login the default user (“linaro”);

Password: linaro

3. Click the **Show Applications** icon  on the bottom left corner to access the system applications;
4. Click **Settings > About** to check more information about the device system.

7.2 System Settings

Upon a click of **Show Applications**, users can click or search a single application for specific use, or modify system settings in **Settings** application.

7.2.1 Language setup

Users may set the display language and language format after accessing **Settings > Region & Language**. With the use of **Language Support**, users can also install/remove input languages. All the modifications will take effect after system reboot.

Commands for language settings are same as those in 6.2.1.

7.2.2 Date & time setup

To change the system date and time, click **Settings > Date & Time** to open the setup page:

1. Click **Unlock** and enter the password before editing;
2. Toggle on **Automatic Date & Time** to keep the device time with the Internet, or toggle off the option and input the date and time manually;
3. Select/search the target time zone and set the time format;
4. Click **Lock** at the top of the interface to save the settings;
5. Exit and await the settings to take effect.

7.2.3 Network/Wi-Fi

The device is configured to have Internet access once it connects to a live Ethernet or joins to a Wi-Fi network. Users can add a VPN network in **Settings > Network** interface.

Make sure you have installed the Wi-Fi antennas before connecting to an existing wireless network.

You can input the following command to check the information of the network interface: `ip link show`.

7.2.4 User

Users will be able to edit the system account and add/delete a user after inputting the password and unlocking the current user.

7.3 Image Flashing on an Ubuntu Host

7.3.1 Prerequisites

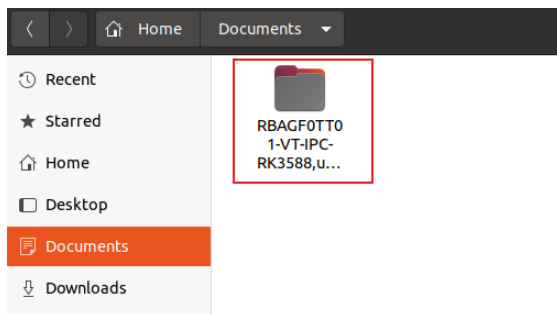
- IBOX3588
- An Ubuntu host (Ubuntu 18.04 or later recommended)
- A USB Type-A to Type-C cable
- Software release package of the device

7.3.2 Image flashing

1. Connect the device with the host computer via the USB Type-A to Type-C cable;
2. Open the Terminal and input the following command line to install ADB tool;

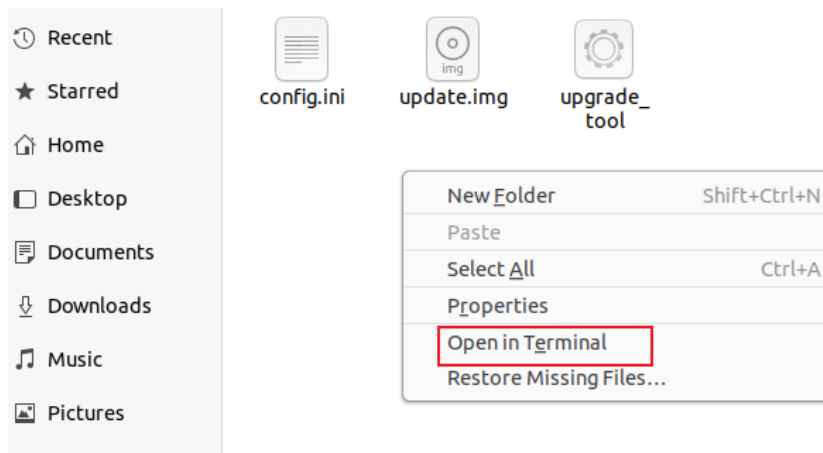
```
$ sudo apt-get install adb -y
```
3. Check if the device is connected to the Ubuntu host via ADB;

```
$ adb devices -l
```
4. Run `adb shell` to execute shell commands on the device;
5. Enter `reboot loader`, and the device will reboot automatically and enter the bootloader mode;
6. Copy the image package provided by Vantron to the Ubuntu host (for instance, in the Documents folder);



7. Unzip the package and open the **img** folder;

8. Right click the mouse in an empty area and click **Open in Terminal** to execute the subsequent commands in the new terminal;



9. Input the following command line to start the upgrade process;

```
sudo ./upgrade_tool uf update.img
```

10. Input the sudo password to download the system image;
11. The system will start upgrading once the download finishes, and the device will reboot automatically when the upgrade finishes.

7.4 ROS Tutorials

Please refer to <http://wiki.ros.org/ROS/Tutorials> for the details of the ROS system.

CHAPTER 8 DISPOSAL AND WARRANTY

8.1 Disposal

When the device comes to end of life, you are suggested to properly dispose of the device for the sake of the environment and safety.

Before you dispose of the device, please back up your data and erase it from the device.

It is recommended that the device is disassembled prior to disposal in conformity with local regulations. Please ensure that the abandoned batteries are disposed of according to local regulations on waste disposal. Do not throw batteries into fire or put in common waste canister as they are explosive. Products or product packages labeled with the sign of “explosive” should not be disposed of like household waste but delivered to specialized electrical & electronic waste recycling/disposal center.

Proper disposal of this sort of waste helps avoid harm and adverse effect upon surroundings and people’s health. Please contact local organizations or recycling/disposal center for more recycling/disposal methods of related products.

8.2 Warranty

Product warranty

VANTRON warrants to its CUSTOMER that the Product manufactured by VANTRON, or its subcontractors will conform strictly to the mutually agreed specifications and be free from defects in workmanship and materials (except that which is furnished by the CUSTOMER) upon shipment from VANTRON. VANTRON's obligation under this warranty is limited to replacing or repairing at its option of the Product which shall, within **24 months** after shipment, effective from invoice date, be returned to VANTRON's factory with transportation fee paid by the CUSTOMER and which shall, after examination, be disclosed to VANTRON's reasonable satisfaction to be thus defective. VANTRON shall bear the transportation fee for the shipment of the Product to the CUSTOMER.

Out-of-Warranty Repair

VANTRON will furnish the repair services for the Product which are out-of-warranty at VANTRON's then-prevailing rates for such services. At customer's request, VANTRON will provide components to the CUSTOMER for non-warranty repair. VANTRON will provide this service as long as the components are available in the market; and the CUSTOMER is requested to place a purchase order up front. Parts repaired will have an extended warranty of 3 months.

Returned Products

Any Product found to be defective and covered under warranty pursuant to Clause above, shall be returned to VANTRON only upon the CUSTOMER's receipt of and with reference to a VANTRON supplied Returned Materials Authorization (RMA) number. VANTRON shall supply a RMA, when required within three (3) working days of request by the CUSTOMER. VANTRON shall submit a new invoice to the CUSTOMER upon shipping of the returned products to the CUSTOMER. Prior to the return of any products by the CUSTOMER due to rejection or warranty defect, the CUSTOMER shall afford VANTRON the opportunity to inspect such products at the CUSTOMER's location and no Product so inspected shall be returned to VANTRON unless the cause for the rejection or defect is determined to be the responsibility of VANTRON. VANTRON shall in turn provide the CUSTOMER turnaround shipment on defective Product within **fourteen (14) working days** upon its receipt at VANTRON. If such turnaround cannot be provided by VANTRON due to causes beyond the control of VANTRON, VANTRON shall document such instances and notify the CUSTOMER immediately.

Appendix Regulatory Compliance Statement

FCC Statement

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

This device complies with Part 15 of FCC Rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) This device must accept any interference received, including interference that may cause undesired operation.

Note: The manufacturer is not responsible for any radio or TV interference caused by unauthorized modifications to this equipment. Such modifications could void the user's authority to operate this equipment.